ALGONQUIN COLLEGE

MODULE 0 : COURSE OUTLINE, REQUIREMENTS, AND RESOURCES

Professor : Dave Houtman

Office: T323

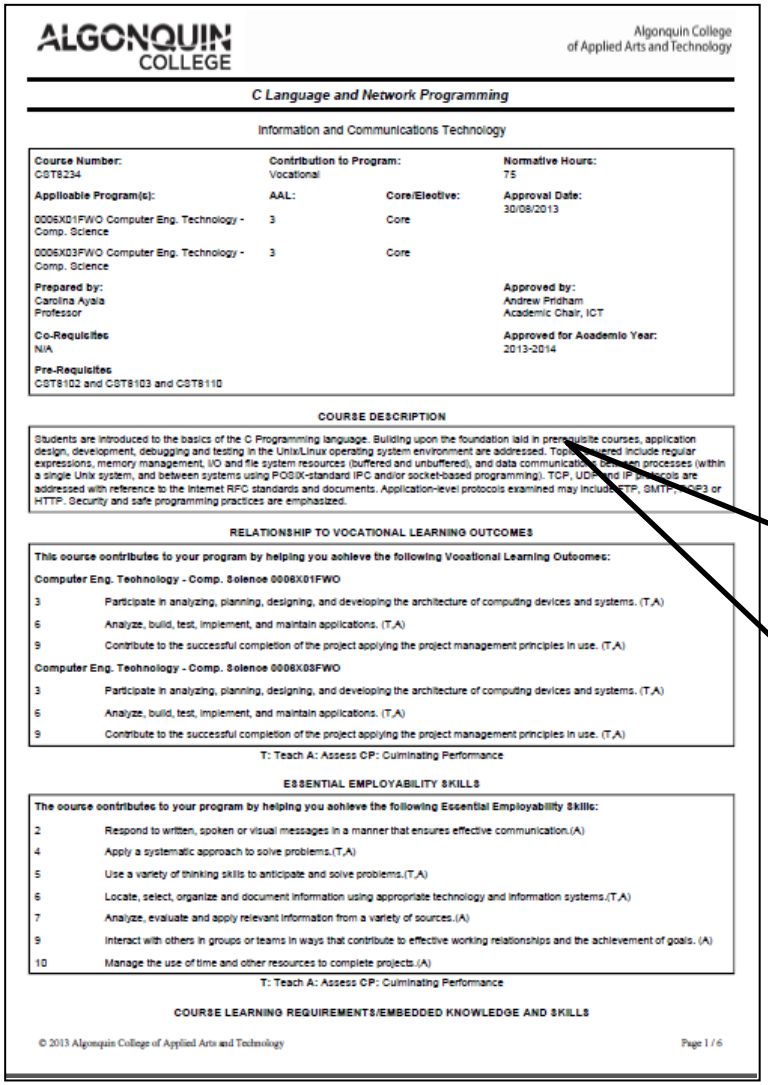
Office Hrs: Monday 15:30 – 16:00

Wednesday 15:30 – 16:00

Friday 15:30 – 16:00

Email: houtmad@algonquincollege.com

Course Outline



ALGONQUIN COLLEGE

Algonquin College of Applied Arts and Technology

C Language and Network Programming

Information and Communications Technology

Course Number:	Contribution to Program:		Normative Hours:
CST8234	Vocational		75
Applicable Program(s):	**AAL:**	**Core/Elective:**	**Approval Date:**
			30/08/2013
0006X01FWO Computer Eng. Technology - Comp. Science	3	Core	
0006X03FWO Computer Eng. Technology - Comp. Science	3	Core	

Prepared by:
Carolina Ayala
Professor

Approved by:
Andrew Pridham
Academic Chair, ICT

Co-Requisites
N/A

Approved for Academic Year:
2013-2014

Pre-Requisites
CST8102 and CST8103 and CST8110

COURSE DESCRIPTION

Students are introduced to the basics of the C Programming language. Building upon the foundation laid in prerequisite courses, application design, development, debugging and testing in the Unix/Linux operating system environment are addressed. Topics covered include regular expressions, memory management, I/O and file system resources (buffered and unbuffered), and data communications between processes (within a single Unix system, and between systems using POSIX-standard IPC and/or socket-based programming). TCP, UDP and IP protocols are addressed with reference to the Internet RFC standards and documents. Application-level protocols examined may include FTP, SMTP, POP3 or HTTP. Security and safe programming practices are emphasized.

RELATIONSHIP TO VOCATIONAL LEARNING OUTCOMES

This course contributes to your program by helping you achieve the following Vocational Learning Outcomes:

Computer Eng. Technology - Comp. Science 0006X01FWO

3	Participate in analyzing, planning, designing, and developing the architecture of computing devices and systems. (T,A)
6	Analyze, build, test, implement, and maintain applications. (T,A)
9	Contribute to the successful completion of the project applying the project management principles in use. (T,A)

Computer Eng. Technology - Comp. Science 0006X03FWO

3	Participate in analyzing, planning, designing, and developing the architecture of computing devices and systems. (T,A)
6	Analyze, build, test, implement, and maintain applications. (T,A)
9	Contribute to the successful completion of the project applying the project management principles in use. (T,A)

T: Teach A: Assess CP: Culminating Performance

ESSENTIAL EMPLOYABILITY SKILLS

The course contributes to your program by helping you achieve the following Essential Employability Skills:

2	Respond to written, spoken or visual messages in a manner that ensures effective communication.(A)
4	Apply a systematic approach to solve problems.(T,A)
5	Use a variety of thinking skills to anticipate and solve problems.(T,A)
6	Locate, select, organize and document information using appropriate technology and information systems.(T,A)
7	Analyze, evaluate and apply relevant information from a variety of sources.(A)
9	Interact with others in groups or teams in ways that contribute to effective working relationships and the achievement of goals. (A)
10	Manage the use of time and other resources to complete projects.(A)

T: Teach A: Assess CP: Culminating Performance

COURSE LEARNING REQUIREMENTS/EMBEDDED KNOWLEDGE AND SKILLS

Students are introduced to the basics of the C Programming language. Building upon the foundation laid in prerequisite courses, application design, development, debugging and testing in the Unix/Linux operating system environment are addressed. Topics covered include regular expressions, memory management, I/O and file system resources (buffered and unbuffered), and data communications between processes (within a single Unix system, and between systems using POSIX-standard IPC and/or socket-based programming). TCP, UDP and IP protocols are addressed with reference to the Internet RFC standards and documents. Application-level protocols examined may include FTP, SMTP, POP3 or HTTP. Security and safe programming practices are emphasized.

Course Description

ALGONQUIN COLLEGE

Algonquin College
of Applied Arts and Technology

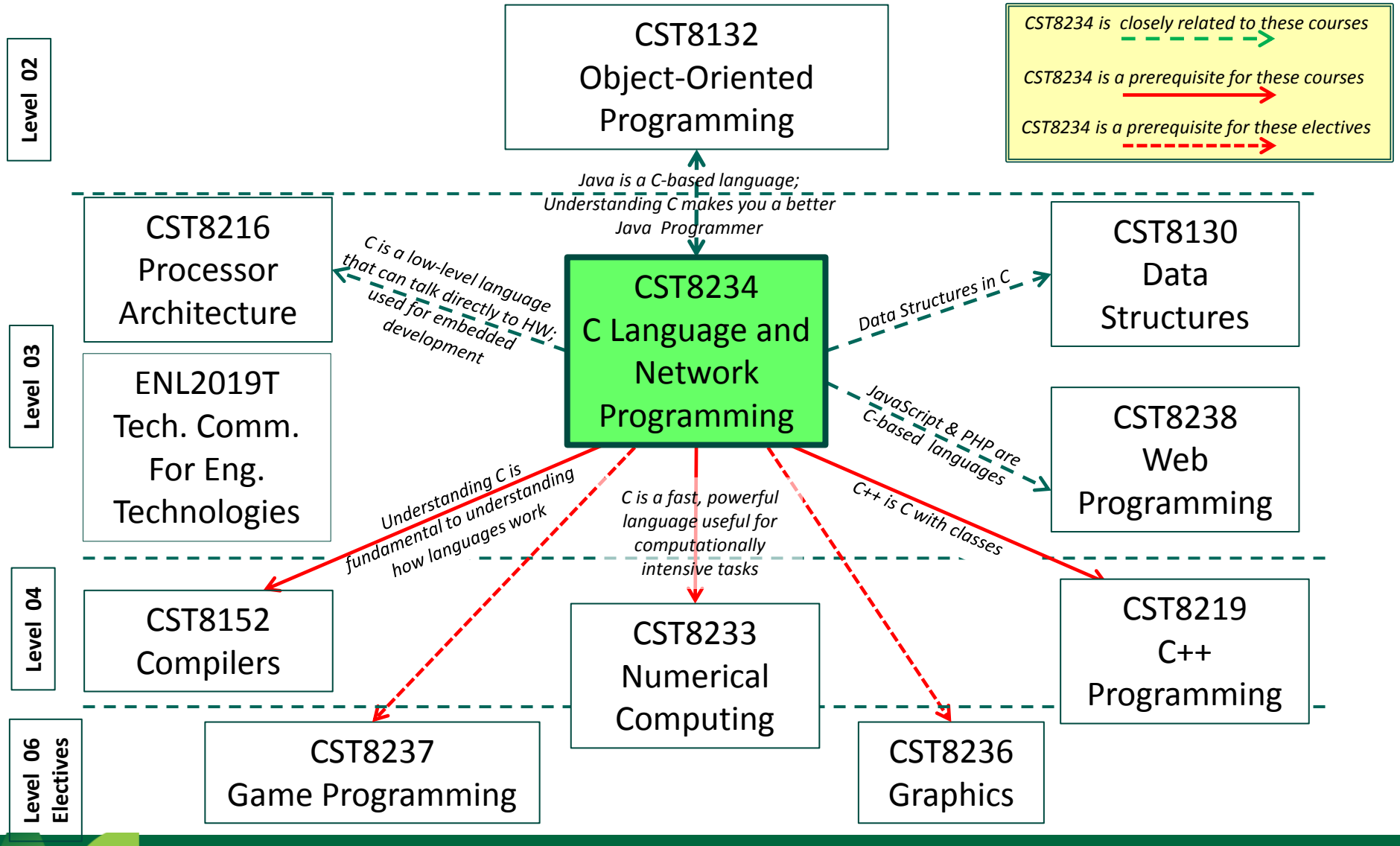
COURSE LEARNING REQUIREMENTS When you have earned credit for this course, you will have demonstrated the ability to:	EMBEDDED KNOWLEDGE AND SKILLS
1. Develop complete C programs using a top-down design and structured programming techniques	• Understand and use strict ANSI C • Use Unix Makefiles to manage complex multi-file source code compilation and linking
2. Learn how to prevent programming errors by planning application programs, and demonstrate the use of both proper design techniques and debugging tools	• Use basic analysis and design principles to describe the problem solution thoroughly, with a suitable test regimen, before beginning to code and test • Make use of diagnostic warning and error messages, inserted output statements, and debugging tools such as gdb and ddd (requires the use and manipulation of numbers in various bases) to correct errors of syntax and logic;
3. Appropriately use static, automatic, and dynamic memory	• Choose and manage memory allocation schemes depending upon each problem's requirements
4. Introduce the concept of pointers and use them in a variety of programming techniques	• Understand the relationship between pointers and arrays • Be able to draw a diagram explaining the concept of pointers and pointee • Be able to use complex data structures using pointers • Understand the concept of passing arguments to a function by value and by reference • Be able to declare and use pointer to functions

Course Description

5. Be able to read, write and update files in C	• Use and understand the standard library I/O functions for stdin, stdout and stderr, as well as other files
6. Know when and how to create and control processes in a Unix/Linux environment	• Create and manage processes with POSIX compliant system calls
7. Illustrate common network protocols of the internet (TCP/IP - UDP)	• Select client and/or server application models as directed to address communication needs; • Make use of standard POSIX-compliant services for socket-based network and inter-process communication services

Note: Despite the name of the course, most of it (approximately 80%) deals with learning to program in C; only ~20% (towards the end of semester) deals with Linux networking. Why? Even though C is similar to Java, there are enough fundamental differences between the two languages that we need to invest a considerable amount of time learning C before we can move on to the networking portion of the course.

How This Course Relates to Your Other Courses



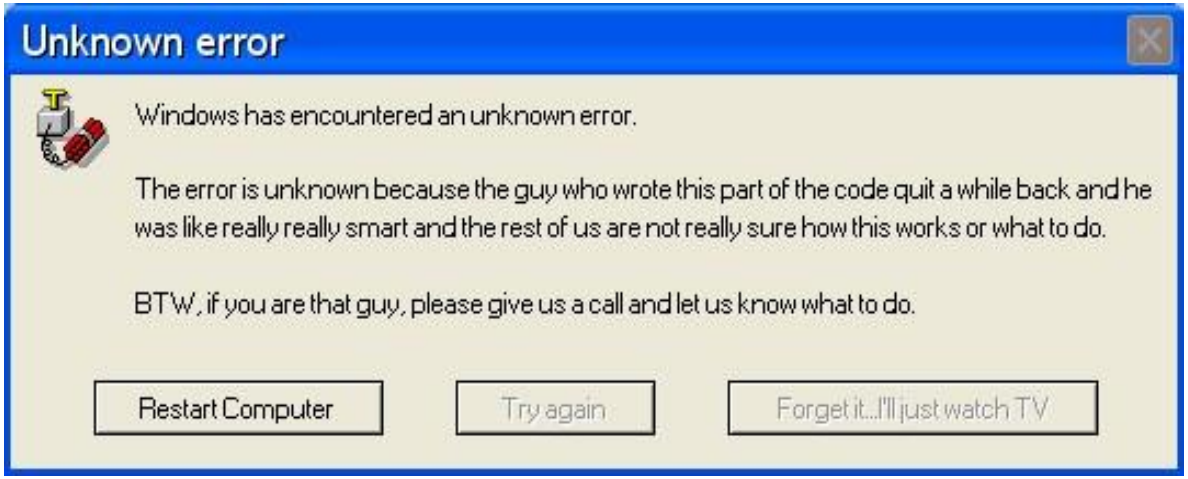
How This Course Enhances Your Programming Skills ⌘

In addition to the specific skills described in the course outline, this course will provide you with the following practical knowledge:

- It expands your understanding of other programming languages (something *different* from Java), and of how computer languages *actually work*; C gives you a fresh perspective on programming;
- It helps you understand the relationship between code and computer architecture, and more importantly, helps you to see *why* a detailed understanding of computer hardware makes you a better programmer;
- It helps you to understand the power (and dangers) of communicating directly with memory using pointers, an insight that aids you in every-thing from database design to embedded systems programming;
- It establishes a foundation for C++ programming.

How This Course Enhances Your Programming Skills ⌘

- Finally: Java, by its nature, tends to encourage students to write large programs, even when these programs are designed to accomplish relatively simple tasks. This is largely a byproduct of the Object-Oriented philosophy, and it is unavoidable. One unfortunate consequence of this is that students often write bloated code, perhaps thinking that this is the norm. While this code often works properly, it will generally:
  - *Be slower to execute;*
  - *Take longer to write and debug than necessary; and*
  - *Be harder to maintain and update by the people responsible*



How This Course Enhances Your Programming Skills ⌘

C, by comparison, is *terse* and, in comparison with Java, *elegant* in its simplicity and compactness. Familiarity with C will introduce you to the virtues of writing smarter, more compact Java code, without the loss of clarity or functionality.

"Everything should be made as simple as possible, but not simpler"

— Einstein

Required Resources

Required Textbook

- The C Programming Language by Brian W. Kernighan & Dennis M. Ritchie, Prentice Hall, Inc., 1988. ISBN 0-13-110362-8

Recommended References*

- The C Programming Language - Exercise Solutions
  http://clc-wiki.net/wiki/K%26R2_solutions
- C: How to Program, 6th Ed.
  Deitel & Deitel ©2010 - Prentice Hall - Published: 10/29/2009
  ISBN-10: 0136123562 ISBN-13: 9780136123569
- A Book on C: Programming in C 4th Ed.
  Al Kelley & Ira Pohl © 1998 – Addison Wesley
  ISBN-10: 0201183994 ISBN-13: 9780201183993

**C has been around for a long time, and there are tons of references available both online and in paper form. The above list represents a tiny fraction of what's available to you.*

Required Resources

Note that the K & R book is a true classic in the world of programming: a copy of this book is an essential item on any programmer's bookshelf. It has sold millions of copies worldwide and has been translated into more than two dozen languages.

Source:http://eqn.princeton.edu/2009/10/kernighan_profile_in_computerworld_is_slashdotted/

bible *n. 1. One of a small number of fundamental books such as* ***Knuth*** *and* ***K&R****. 2. The most detailed and authoritative reference for a particular language, operating system, or other complex software system.*

- E.S. Raymond, The New Hacker's Dictionary, pg. 59

Required Resources

Computer Requirements:

- This program is a *mandatory* laptop program (no exceptions) – see the specifications at http://mlearning.algonquincollege.com
- Use Non-Windows-based laptops at your own risk

Software Requirements:

- VMWare (should be already installed on your PC)
- Ubuntu 12.04 required for this course, plus
  - gcc
  - OpenGL
  - ddd, gdb

  which will be (loaded in Lab 1)

Teaching / Learning Methods

Lectures

- 3 hours in class per week; some (but not all) notes/materials will be posted online.
- Attendance will be taken each class (for information purposes only)

Labs

- 2 hours per week in lab with computers
- Labs and Assignments will be posted online

Course Evaluation

Theory — 55%

- Midterms (2 total – Oct. 1 and Nov. 5) — **20%**
- Quizzes (Best 10 out of 11, worth 1% ea., approx. 1 ea. week) — **10%**
- Final Exam — **25%**

Practical — 45%

- Labs/Assignments — **30%**
  - Completion of basic lab work (~ 8 short labs expected) — **10%**
  - Assignments (3 – 4 expected) — **20%**

- Skilled-Based Assessment (SBA) — **15%**
  - 3 total, worth 5% each

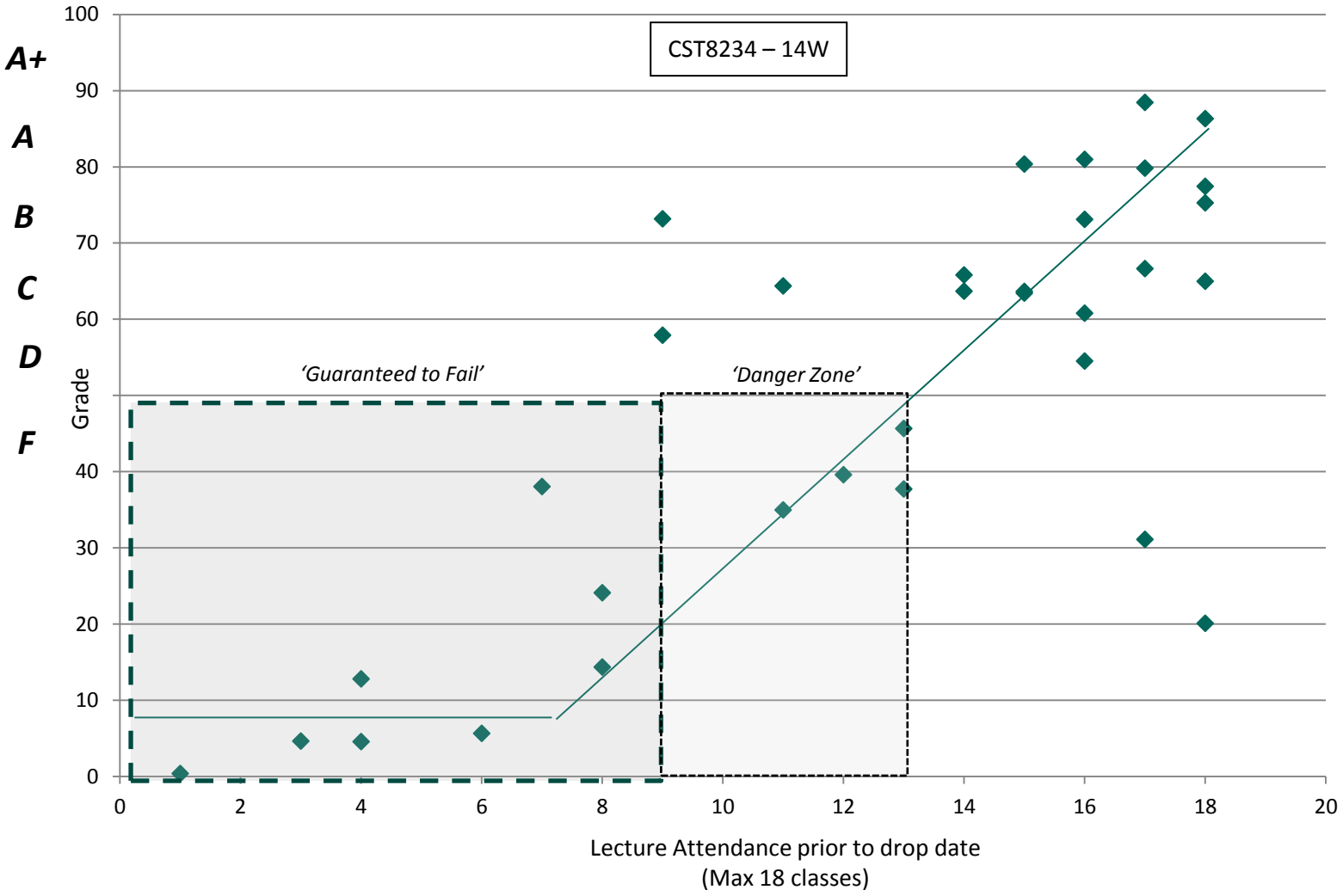
Note that you must pass more than 50% on the theory portion in order to pass this course, i.e. 27.5%/55.0%.

Getting the Most Out of This Course

Many of the marks lost by students during the semester are due not to a failure to comprehend course material (resulting in poor marks on exams and assignments), but due to simple negligence. Marks creep away a few percent at a time—1% for each missed weekly quiz, 5% for missed SBAs, etc.—and the A- or B+ that a student might have had otherwise, if properly attending to the details of the course, quickly slips away until by final exam time the student is scrambling to obtain a passing grade. To avoid this situation:

1. Stay on top of things; don't put off deadlines to the last minute;
2. Anticipate the busy times of the semester and try to get ahead of schedule beforehand, whenever possible;
3. Practice good time management skills. Plan your day *at the start of it* by setting realistic goals to be achieved by the end of it;
4. A good night's sleep is more important than a relaxing evening spent 'just vegetating';
5. **Don't miss labs. Don't miss lectures**—this is where good grades quickly slip away.

How Attendance Affects Your Grade



Student's Academic Responsibilities

Each student is responsible for:

- ***Knowing the due dates*** for assignments and exams.
- ***Attending*** all classes. Anything covered in class, whether in the slides or covered verbally in lectures, is fair game for tests, quizzes, etc.
- ***Completing course work on time.***
- ***Maintaining*** a folder of all work done in the course during the semester for validation claims in cases of disagreement with faculty.
- ***Keeping both paper and electronic copies of all assignments***, marked and unmarked, in case papers are lost or go missing;
- ***Regularly checking*** your Algonquin e-mail account for important messages from both professors and college administration.
- ***Staying on top of the course material each week*** throughout the semester. During those weeks when there's no midterm exam, there's always a quiz, so stay don't wait until exam time to start studying.

Plagiarism

- **The School of Advanced Technology's Standard Operating Procedure on Plagiarism and Academic Honesty** defines plagiarism as an attempt to use or pass off as one's own idea or product, work of another without giving credit. Plagiarism has occurred in instances where a student either directly copies another person's work without acknowledgement; or, closely paraphrases the equivalent of a short paragraph or more without acknowledgement; or, borrows, without acknowledgement, any ideas in a clear and recognizable form in such a way as to present them as one's own thought, where such ideas, if they were the student's own would contribute to the merit of his or her own work.

- Plagiarism is one of the most serious academic offences a student can commit. Anyone found guilty will, on the first offence, be given a written warning and an "F" on the plagiarized work. If the student commits a second offence, an "F" will be given for the course along with a written warning. A third offence will result in suspension from the program and/or the college.

Miscellaneous

Missed Classes due to Illness

- As per College policy, students are not required to supply a doctor's letter if they are absent due to a cold or flu, and miss a class or an exam as a result. However, students should send me an email (houtmad@algonquincollege.com) as soon as they become aware that they will be missing a class. *However*, if you are sick for more than 48 hours and miss more than two consecutive labs/lectures, you should plan to supply a doctor's letter when you return to class. (Note: if you visit the doctor due to illness, be sure to pick up a letter, even if you don't think your illness will impact your attendance.)
- If you miss an exam as a result of illness, the standard procedure is to pro-rate your remaining exams to account for the missed mark.
- Since students will generally have two weeks or more to do labs, sickness is generally not an excuse for submitting a lab late, unless it is due to a prolonged illness, in which case a doctor's letter will be required.

CSD Students

- Students who require special consideration during exams, as recognized by CSD, should present me with the appropriate paperwork as soon as possible so that appropriate accommodation can be reached ahead of time.

Miscellaneous

Late Assignments and Labs

- Completed labs should be demonstrated on or before the stated deadline during your lab period; in cases in which this is not possible due to either technical problems or time limitations at the end of the lab period, labs are due no later than the end of the week (you can drop by during office hours and demonstrate your lab then, if necessary). Under special circumstances, the lab submission deadline may be extended into the next week, but in practice this rarely happens.

- Late assignments will be penalized as follows:

> 1 minute late to <= 10 minutes late	=	10.0% off your mark
> 10 minutes late to *<= 6 hours late*	=	20.0% off your mark
> 6 hours late to <= 12 hours late	=	33.3% off your mark
> 12 hours late to <= 18 hours late	=	50.0% off your mark
> 18 hours late	=	Ha! Don't even bother…

  Again, extensions are possible under special circumstances, but unlikely.

- If you make multiple submissions of the same assignment, only the last assignment is counted.

Miscellaneous

Distractions

- Students who engage in activities which are distracting will be asked to leave. 'Distractions' includes:
  - Any conversation with a classmate above a whisper
  - Displaying any material on their PC which is not related to your coursework. This includes, in particular, playing videos during class time.

Electronic Devices

- Turn off all cell phones before class (and anything else that makes a noise, for that matter)

eCigarettes

- are prohibited during class and during lectures.

Miscellaneous

~~Blackboard~~ Canvas

- Make sure you can select CST8234_400 from the menu
- The web site contains the following folders:
  - Course Outline
  - Course Lecture Notes
  - Labs
  - Assignments
  - Textbook & Resources
  - Published Solutions to Tests, Quizzes, and Assignments
  - Announcements and deadlines
  - Grades
  - Drop box for assignments

This is a 'Middle Earth Battlefield Simulator' Free Zone

The Middle Earth Humane Society has certified that no orcs, dwarves, elves, nazgul, hobbits, humans, wizards, trolls, goblins, balrogs, eagles, ents, giant spiders, oliphants, trolls, dragons, wargs, or dark lords were harmed in the making of this course. It is also Certified Organic. And nut-free, for those of you with allergies.

Weekly Schedule* - CST8234

Time	Mon	Tues	Wed	Thurs	Fri
14:00-15:00					
15:00-16:00	Office Hrs 15:30-16:00		Office Hrs 15:30-16:00		Office Hrs 15:30-16:00
16:00-17:00	CST8234 Sec. 011 Lab B110		CST8234 Sec. 010 Lecture T117		CST8234 Sec. 013 Lab T126
17:00-18:00					
18:00-19:00		CST8234 Sec. 012 Lab T111		CST8234 Sec. 010 Lecture T130	
19:00-20:00					

*Office Hours subject to possible change, depending on circumstances, scheduling conflicts, outside obligations, etc…but hopefully stable for the first few weeks of the semester.

Exam Schedule \ Important Deadlines & Dates

September				
Monday	Tuesday	Wednesday	Thursday	Friday
1 **Labour Day**	2 **First Day of Class**	3 **First Lecture**	4	5
8	9	10 **Quiz #1 – 1%**	11	12
15 ***Early With-drawal Date***	16	17 **Quiz #2 – 1%**	18	19
22	23	24 **Quiz #3 – 1%**	25	26
29	30			

October				
Monday	Tuesday	Wednesday	Thursday	Friday
		1 **Midterm#1–10%**	2	3 **SBA#1 – 5%**
6 **SBA#1 – 5%**	7 **SBA#1 – 5%**	8 **Quiz #4 – 1%**	9	10
13 **Thanksgiving**	14	15 **Quiz #5 – 1%**	16	17
20	21	22 **Quiz #6 – 1%**	23	24
27	*28* **SBA#2 – 5%**	29 **Quiz #7 – 1%**	30	31 **SBA#2 – 5%**

Exam Schedule \ Important Deadlines & Dates

November				
Monday	Tuesday	Wednesday	Thursday	Friday
3 SBA#2 – 5%	4	5 Midterm#2–10%	6	7 *Final Drop Date – your last chance for a 'W'*
10	11	12 Quiz #8 – 1%	13	14
17	18	19 Quiz #9 – 1%	20	21
24 SBA#3 – 5%	25 SBA#3 – 5%	26 Quiz #10 – 1%	27	28 SBA#3 – 5%

December				
Monday	Tuesday	Wednesday	Thursday	Friday
1 Open Lab	2 Open Lab	3 Quiz #11 – 1%	4 Last Lecture	5 Open Lab
8	9	*Exam Week – Final Exam Date TBA*		12